1. 什么是线性回归模型？它的基本形式是什么？

答案：

线性回归模型是一种用于预测连续变量的统计方法，它假设自变量与因变量之间存在线性关系。其基本形式为：

y = w₁x₁ + w₂x₂ + ... +wₙxₙ + b

其中，w 是权重，b 是偏置，x 是输入变量，y 是预测值。

4. Optimization

- Preprocess: incorporate the bias w_0 into **w** by using $x_0 = 1$ (Add an **1** to input **x**) . Then, $\mathbf{x} = \begin{bmatrix} 1 \\ x \end{bmatrix}$
- Linear regression model: $y(x) = w^T\mathbf{x}$
- MSE loss: $l(w) = \frac{1}{2N}\sum_{n=1}^{N}[t^{(n)} - y(x^{(n)})]^2$, convex

4.1 Least square solution

i. let the gradient equal to 0, to find the minima: $\nabla l(w) = -\frac{1}{N}\sum_{n=1}^{N}(t^{(n)} - w^T\mathbf{x}(n))\mathbf{x}^{(n)} = 0$

ii. then we get: $w = (\mathbf{x}^T\mathbf{x})^{-1}\mathbf{x}^T t$

4.2 Gradient decent

- let gradient decrease to the smallest through iteration: Initailize at one point, calculate its gradient and move in the opposite direction.
- Protocol:
  a. initialize w (randomly)
  b. repeatedly update w based on the gradient, λ is the learning rate

2. 感知机模型的原理是什么？它的主要应用是什么？

答案：

感知机模型是一种用于二分类问题的线性分类器，它基于输入特征的加权和并通过激活函数（符号函数）进行分类。其主要应用是解决线性可分问题。

感知机模型的输出：

f(x) = sign(w * x + b)

若 f(x) > 0，分类为正类；否则为负类。

11. Write down the gradient-descent law and gradient-descent law with momentum, respectively.

Gradient Descent Update Rule :

$$\theta_{t+1} = \theta_t - \eta\nabla_\theta L(\theta_t)$$

- θ represents the parameters of the model.
- L is the loss function.
- $\nabla_\theta L(\theta_t)$is the gradient of the loss function with respect to the parameters θ at iteration t.
- η is the learning rate, a scalar that determines the size of the step to take on each iteration.

Gradient Descent with Momentum

$$v_{t+1} = \gamma v_t + \eta\nabla_\theta L(\theta_t)$$
$$\theta_{t+1} = \theta_t - v_{t+1}$$

- v_t is the velocity at time step t.
- γ is the momentum coefficient.

7. Normalization

- Improving model accuracy: Comparability in values between features across different dimensions can significantly enhance the accuracy of model learning.
- Accelerating learning convergence: Searching for the optimum becomes notably smoother, making it easier for the model to converge correctly to the optimal solution.

7.1 Min-Max normalization

$$x^* = \frac{x - x_{min}}{x_{max} - x_{min}}$$

- Maps the data along any dimension to [0,1]
- The purpose of **min-max normalization**: make the impact of each feature compatible, which involves scaling transformations of the features
- Normalizing data will alter the distribution of the feature data.

3. 逻辑回归模型与线性回归的区别是什么？

答案：

逻辑回归模型用于解决分类问题，而线性回归用于预测连续变量。逻辑回归通过 sigmoid 激活函数将线性回归的输出转换为概率值，其形式为：

p(y=1|x) = 1/(1+e^(-(w*x+b)))

其中 p 是事件发生的概率。

4. SGD，BGD，MBGD 算法分别是什么？它们的意义分别是什么？

特性	BGD	SGD	MBGD
每次使用样本数	全部样本	单个样本	小批量样本
计算效率	慢	快	介于两者之间
梯度方向稳定性	稳定	噪声较大	相对稳定
适合大数据集	不适合	非常适合	适合
适合在线学习	不适合	非常适合	较适合
易受局部最优影响	容易受影响	噪声帮助跳出局部最优	较难受影响

7.2 Mean normalization

$$x^* = \frac{x - \mu}{\sigma}$$

where

$$\mu = \frac{1}{N}\sum_{n=1}^{N}x^{(i)}, \sigma^2 = \frac{1}{N}\sum_{n=1}^{N}(x^{(i)} - \mu)^2$$

- The data becomes **zero mean** and **unit variance**
- **Mean normalization**: aims to make different features comparable to each other
- The distribution of the feature data remains unchanged

梯度下降避免陷入非最小值的极小值中的方法：

学习率调整：适当调整学习率可以帮助避免陷入极小值。如果学习率过小，收敛速度会很慢；如果过大，可能会越过最小值。

动量（Momentum）：在梯度下降中加入动量项可以帮助加速收敛，并且能够越过一些小的极小值。

自适应学习率方法：如AdaGrad、RMSProp和Adam等优化算法，它们能够根据参数的更新历史自适应地调整学习率。

使用更高级的优化算法：如共轭梯度法、拟牛顿法（BFGS）等，这些算法在寻找最小值时更加有效。

正则化：通过添加正则化项（如L1或L2正则化）可以减少过拟合的风险，并且有助于避免陷入局部最小值。

多次初始化：从不同的初始点开始多次运行梯度下降算法，选择最好的结果。

使用全局优化方法：如模拟退火、遗传算法等。这些方法在理论上可以找到全局最小值

问题 2：感知机模型的决策边界是什么？

- **答案**：感知机模型的决策边界是一个线性超平面，由方程 $w_0 + w_1x_1 + \ldots + w_dx_d = 0$ 定义。

问题 3：逻辑回归模型如何处理类别概率？

- **答案**：逻辑回归模型使用sigmoid函数 $g(z) = \frac{1}{1+\exp(-z)}$ 将线性函数的输出映射到概率值，从而处理类别概率。

10. In a perceptron, what happens if the data are not linearly separable?
A. The perceptron will still converge to a solution.
B. The perceptron algorithm will not converge.
C. The perceptron automatically switches to a non-linear model.
D. The perceptron will discard non-separable points.
Correct Answer: A

It will still converge, but the loss value is relatively large.

Logistic Regression 相对于 Perceptron 模型的优点主要包括：

- 概率解释：Logistic Regression 提供了概率输出，其输出值介于0和1之间，可以被解释为样本属于正类的概率。这使得我们能够了解预测的置信度，从而进行更深入的分析和应用。
- 良好的可解释性：Logistic Regression 的结果容易解释，模型系数可以直接理解为独立变量对发生依赖变量概率的影响强度
- 稳健性：与其他回归模型相比，Logistic Regression 对异常值的敏感性较低，并且能够处理变量之间的非线性关系
- 可扩展性：得益于现代计算工具，Logistic Regression 可以应用于大型数据集，适合大数据应用和复杂研究领域，如基因组学和计量经济学
- 多类别分类：Logistic Regression 可以很容易地扩展到多类别分类问题，通过使用softmax分类器，这被称为多项式Logistic Regression

题目: k折交叉验证（k-fold cross-validation）的原理是什么？请简要描述其步骤。

答案: k折交叉验证是一种评估模型性能的方法，其原理是将训练数据集随机分成k个大小相等的子集。然后，依次选择其中一个子集作为验证集，其余k-1个子集作为训练集，进行k次训练和验证。最后，将k次验证结果的平均值作为模型的最终性能评估。具体步骤如下：

1. 将训练数据集分成k个子集。
2. 对于每个子集，使用该子集作为验证集，其余子集作为训练集。
3. 训练模型并计算验证集上的性能。
4. 重复步骤2和3，直到所有子集都被用作验证集。
5. 计算k次验证结果的平均值。

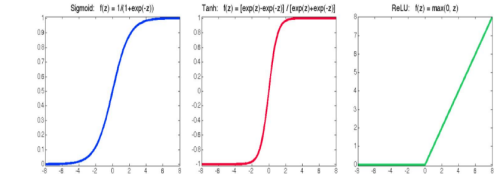
20. What is differences between linear regression and logistic regression？ Accuracy: 88/99

- Outcome Variable:
  Linear Regression: Used for predicting **a continuous outcome** variable (dependent variable).
  Logistic Regression: Used for predicting **a categorical outcome variable.**
- Loss Function:
  Linear Regression: Typically uses **Mean Squared Error** (MSE).
  Logistic Regression: **binary cross-entropy**.
- Assumptions:
  Linear Regression: The relationship between the independent and dependent variables is linear. Observations are independent of each other. The residuals (errors) of the model are normally distributed.
  Logistic Regression: The dependent variable is binary. The logit transformation (log-odds) of the probability is a linear combination of the predictors.

2.2.1 Concept 1: Page 29, Meaning of accuracy, recall, precision and F1 score

In the context of the accuracy, recall, precision and F1 score, there are only the expression of them, but lack the description of what they mean:
1. Accuracy: Represents the percentage of all forecasts that are correct and measures overall performance
2. Recall: Represents the proportion of all actual positive classes that are correctly identified, focusing on the omission rate
3. Precision: Represents the proportion of all predicted positive classes that are actually positive, focusing on the false positive rate
4. F1 score: Represents the harmonic average of accuracy and recall, weighing the performance between the two

- Activation functions:
  - **Sigmoid**: $\sigma(x) = \frac{1}{1+e^{-z}}$
  - **Tanh**: $tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$
  - **ReLU(Rectified Linear Unit)** $ReLU(z) = max(0, z)$



- Neural network architecture
  - Naming convention: $N_{layer} = (N-1)$(layers of hidden units) $+1$(output layer), input layer is not counted

oid

点

- 数学表达式：$\sigma(x) = \frac{1}{1+e^{-z}}$
- 输出范围：(0, 1)
- 形状：S形曲线
- 优点：输出值被压缩在0和1之间，可以将输入映射到一个概率分布上，因此在二分类问题中应用广泛。
- 缺点：容易出现梯度消失问题，特别是在深层网络中，因为其导数在输入值较大或较小时接近于0，导致梯度更新非常缓慢。此外，它不是零中心化的，即输出值的均值不是0，这可能会导致梯度更新方向不稳定。

适用情况

- 主要用于二分类问题，例如逻辑回归模型中。

ReLU

- 特点
  - 数学表达式：$\mathrm{ReLU}(x) = \max(0, x)$
  - 输出范围：(0, +∞)
  - 形状：折线形
  - 优点：计算简单，收敛速度快，能够有效缓解梯度消失问题，因为其导数在正区间内为常数1。此外，它具有稀疏激活特性，即大部分神经元输出为0，这有助于提高模型的计算效率和泛化能力。
  - 缺点：存在"死亡ReLU"问题，即当输入值小于0时，输出和梯度都为0，导致这部分神经元不再更新，从而失去作用。此外，输出不是零中心化的。
- 适用情况
  - 适用于深层神经网络中的隐藏层，因为其计算效率高且能够缓解梯度消失问题。在卷积神经网络(CNN)和循环神经网络(RNN)等模型中应用广泛。

Tanh

- 特点
  - 数学表达式：$\tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$
  - 输出范围：(-1, 1)
  - 形状：S形曲线
  - 优点：输出是零中心化的，即均值为0，这有助于梯度更新方向稳定。相比于sigmoid，其输出范围更大，因此在某些情况下可以提供更好的梯度传播。
  - 缺点：同样容易出现梯度消失问题，特别是在深层网络中，因为其导数在输入值较大或较小时也接近于0，导致梯度更新缓慢。此外，计算复杂度比ReLU高。
- 适用情况
  - 可以用于一些需要零中心化输出的神经网络层，例如在某些循环神经网络(RNN)变体中，如LSTM和GRU的某些部分。但在深层网络中，由于梯度消失问题，其应用受到一定限制。

17. What does an MLP with one hidden layer theoretically capable of representing?
A. Only linear relationships
B. Any continuous function, acting as a universal approximator.
C. Only binary classification functions.
D. Functions that require at least three layers to compute.
Correct Answer: B

4. 什么是多层感知机（MLP）神经网络模型？它的结构有哪些组成部分？

答案：

多层感知机（MLP）是包含一个或多个隐藏层的前馈神经网络，能够解决非线性问题。其结构包括：

输入层：接收输入数据。

隐藏层：通过激活函数进行特征学习。

输出层：产生最终结果。

MLP 通过反向传播算法进行训练，更新权重和偏置。

b. Backward propagation:

$$\delta_k^o = \frac{t_k - o_k}{o_k(1-o_k)}, \delta_k^z = \delta_k^o \cdot o_k(1-o_k), \frac{\partial E}{\partial w_{kj}} = \delta_k^z \cdot h_j$$

$$\delta_j^h = \sum_k \delta_k^z w_{kj}, \delta_j^u = \delta_j^h \cdot f'(u_j)$$

$$\frac{\partial E}{\partial v_{ji}} = \delta_j^u x_i$$

Forward propagation:

$$u_j = v_{j0} + \sum_{i=1}^{D} v_{ji}x_i, h_j = f(u_j), z_k = w_{k0} + \sum_{j=1}^{D} w_{kj}h_j, o_k = g(z_k)$$

w(15+20+8)+b(5+4+2)=54

2. Suppose an MLP model has 3 input nodes, 2 hidden layers, 5 nodes in the first layer, 4 nodes in the second layer, and 2 nodes in the output layer. All layers are fully connected. Calculate the total number of learnable parameters in the network.

12. How can we check overfitting? List all approaches for avoiding overfitting. Accuracy: 48/99

Get any one points

Check overfitting:

- **Training Accuracy but Low Validation/Test Accuracy**: If the model performs exceptionally well on the training data but poorly on the validation/test data, it is likely overfitting.
- Plotting training and validation loss during training can show if the training **loss continues to decrease while the validation loss starts to increase**, indicating overfitting.
- **Cross-validation**

Get any four points

Approaches for avoiding overfitting:

- **Simplify the Model**
- **L1 and L2 Regularization**
- **Dropout**
- **Early Stopping**
- **Data Augmentation**
- **Increase Training Data**

18. In an MLP, what is the role of the activation function?
A. To linearly combine inputs from the previous layer.
B. To add bias to each neuron's output.
C. To introduce non-linearity into the model.
D. To reduce the dimensionality of the input data.
Correct Answer: C

Question6: When changing MLP from binary to multiclass, what should be changed?

Answer:

- The activation function should be changed from sigmoid to softmax.

1. Regression
Mean Squared Error (MSE)

$$L(y, \hat{y}) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

Mean Absolute Error (MAE)

$$L(y, \hat{y}) = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

2. Binary Classification
Binary Cross-Entropy Loss

$$L(y, \hat{y}) = -\frac{1}{n}\sum_{i=1}^{n}[y_i \log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)]$$

3. Multi-class Classification
Categorical Cross-Entropy Loss

$$L(y, \hat{y}) = -\sum_{i=1}^{n}\sum_{c=1}^{C}y_{i,c}\log(\hat{y}_{i,c})$$

2. Nearest Neighbors

- Training example in Euclidean space: $x \in R^d$
- Distance typically defined to be Euclidean:

$$\|x^{(a)} - x^{(b)}\|_2 = \sqrt{\sum_{j=1}^{d} (x_j^{(a)} - x_j^{(b)})^2}$$

- Decision boundaries
  - NN does not explicitly compute decision boundaries but can be inferred
  - **Voronoi diagram**

1. Given Training Samples: $\{(1,2),(3,4),(5,6),(7,8)\}$ Corresponding class labels are $\{A,A,B,B\}$. Test sample is $(4,5)$. Using $k=3$ for KNN classification and Euclidean distance as the metric, predict its class label.

   Answer: - Distance Calculation:
   $\|(4,5)-(1,2)\| = \sqrt{(4-1)^2+(5-2)^2} = \sqrt{9+9} = \sqrt{18}$
   $\|(4,5)-(3,4)\| = \sqrt{(4-3)^2+(5-4)^2} = \sqrt{1+1} = \sqrt{2}$
   $\|(4,5)-(5,6)\| = \sqrt{(4-5)^2+(5-6)^2} = \sqrt{1+1} = \sqrt{2}$
   $\|(4,5)-(7,8)\| = \sqrt{(4-7)^2+(5-8)^2} = \sqrt{9+9} = \sqrt{18}$

   Nearest neighbors are $(3,4),(5,6),(1,2)$, whose corresponding labels are $\{A,B,A\}$.

   Majority voting predicts the class label as A.

5. KNN 模型的工作原理是什么？

答案：

KNN 模型是一种基于实例的监督学习算法，它根据距离度量选择离目标样本最近的 K 居，进行分类或回归。

分类：通过多数投票法决定类别。

回归：通过邻居的平均值或加权平均值进行预测。

- Rules to choose k
  - larger k may lead to better performance
  - but too large k may be far away from the quety
  - use cross-validation to find k
  - Rule of tumb is $k < \sqrt{n}$, where n is the number of training examples

- smooth when disturbed by mis-labeled data("class noise")
- Algorithm
  - Find k examples $\{\mathbf{x}^{(i)}, t^{(i)}\}$ close to the test instance **x**
  - Classification output is majority class

$$y = argmax_{t^{(z)}} \sum_{r=1}^{k} \delta(t^{(z)}, t^{(r)})$$

- k-NN naturally forms complex decision boundaries and adapts to data density
- k-NN typically works well with lots of samples

- Problems:
  - Sensitive to class noise
  - Sensitive to scales of attributes
  - Distances are less meaningful in high dimensions
  - Scales linearly with number of examples

题目: 决策树与k-NN的主要区别是什么？

答案:

- **决策树**:
  - 决策边界是轴对齐的，树结构。
  - 测试复杂度取决于属性和分裂。
- **k-NN**:
  - 决策边界是分段线性的。
  - 测试复杂度是非参数的，除了训练样本外几乎没有参数。

5. Selecting the right threshold

- Step 1: Sort the Data
  - First, sort the data based on the values of the continuous attribute.
- Step 2: Calculate All Possible Thresholds
  - For the sorted data, thresholds can be taken as the midpoint between any twoconsecutive values. For example, if the sorted data is [1,2, 3, 4, 5], then thepossible thresholds are 1.5, 2.5, 3.5, and 4.5.
- Step 3: Calculate the Information Gain for Each Threshold
  - For each possible threshold, split the data into two parts: values less than orequal to the threshold and values greater than the threshold. Then, calculate theInformation Gain for each split.
- Step 4: Choose the Threshold with Maximum Information Gain
  - From all possible thresholds, select the one with the maximum Information Gainas the final threshold.

6. 决策树模型的核心思想是什么？常用的决策树算法有哪些？

答案：

决策树通过将数据集分割成不同子集，形成一棵树结构，以实现分类或回归。其核心思想是通过特征选择最大化信息增益或减少不纯度。

常用的决策树算法包括：

ID3：基于信息增益。

C4.5：基于信息增益率。

CART：基于基尼指数。

- Entropy H: $H(X) = -\sum_{x \in X} p(x) log_2 p(x)$

$$H(Y|X) = \sum_{x \in X} p(x) H(Y|X=x)$$
$$= -\sum_{x \in X} \sum_{y \in Y} p(x,y) \log_2 p(y|x)$$

题目: 决策树的内部节点和叶子节点分别代表什么？

答案: 内部节点代表属性测试，分支代表属性的不同取值；叶子节点代表最终的类别或输出值

3. Standard PCA

- **Algorithm**: to find M components underlying D-dimensional data

1. Select the top M eigenvectors of C (data covariance matrix):

$$C = \frac{1}{N} \sum_{n=1}^{N} (x^{(n)} - \bar{x})(x^{(n)} - \bar{x})^T = U \Sigma U^T \approx U_{1:M} \Sigma_{1:M} U_{1:M}^T$$

where U is orthogonal, columns are unit-length eigenvectors

$$U^T U = U U^T = I$$

and Σ is a matrix with eigenvalues on the diagonal, representing the variance in the direction of each eigenvector.
Matrix form: $C = \frac{1}{N}(X - \bar{x})(X - \bar{x})^T$
2. Project each input vector x into this subspace, e.g.,

$$z_j = u_j^T x; \qquad z = U_{1:M}^T x$$

Let $U_{pca} = U_{1:M}$. Then we have the principal components

$$z = U_{pca}^T x$$

5. Autoencoders

- a neural network whose outputs are its own inputs (to minimize reconstruction error)
- **Define**

$$z = f(Wx); \quad \hat{x} = g(Vz)$$

- **Goal:**

$$\min_{W,V} \frac{1}{2N} \sum_{n=1}^{N} \left\| x^{(n)} - \hat{x}^{(n)} \right\|^2$$

- If g and f are linear

$$\min_{W,V} \frac{1}{2N} \sum_{n=1}^{N} \left\| x^{(n)} - VWx^{(n)} \right\|^2$$

In other words, the optimal solution is **PCA** for the case when the mean of the data is 0.

- if $g()$ is not linear $\rightarrow$ nonlinear PCA

Question7: What is the difference between PCA and autoencoder?

Answer:

- PCA
  - Linear
  - Use linear transformation to reduce the dimension of data
- Autoencoder
  - Nonlinear
  - Combine PCA and MLP

3. PCA公式

PCA的核心是通过特征值分解来减少数据维度，保留最大方差。其步骤如下：

1. 计算数据的协方差矩阵 $\mathbf{C} = \frac{1}{n}\mathbf{X}^T\mathbf{X}$。
2. 对协方差矩阵进行特征值分解，得到特征值和特征向量。
3. 选择前k个最大特征值对应的特征向量，这些向量构成新的坐标轴，投影数据到这些坐标轴上。

1. What does the first principal component of PCA correspond to in the covariance matrix? Answer: Eigenvector corresponding to the largest eigenvalue

1. What is the main improvement of K-means++ compare to K-means?

   Answer: K-means++ initializes the cluster centers by a probability distribution, so that the distance between the selected initial cluster centers is as large as possible.

Question1:What is the difference between supervised learning and unsupervised learning?

Answer:

- Supervised learning:
  - Require a training set of example inputs
  - Require a set of desired outputs
  - Genarate output for new inputs
- Unsupervised learning:
  - Do not require a training set
  - Do not require a set of desired outputs
  - Detect patterns in data

Question3:What is the steps of K-means clustering?

Answer:

- Step1: Randomly initialize k cluster centers
- Step2: Calculate the distance between each point and the k cluster centers
- Step3: Assign each point to the closest cluster center
- Step4: Update the cluster centers
- Step5: Repeat step2-4 until the cluster centers do not change

What is the main difference between Soft K-means and Hard K-means?
A. Soft clustering uses a different distance metric
B. soft clustering always results in fewer clusters than hard clustering
C. hard clustering uses probabilities to update the cluster centers
D. soft clustering allows data points to belong to multiple clusters with different probabilities

The K-means algorithm may not perform well with which of the following types of data?
A. convex and isotropic data
B. non-convex or anisotropic data
C. uniformly distributed data
D. linearly differentiable data

什么是支持向量？
支持向量是离超平面最近的点，它们决定了超平面的方向和位置。这些点是 SVM 的关键，因为其他样本点不会影响超平面的优化。

什么是核函数（Kernel Function）？有哪些常见的核函数?
核函数是一种用于将低维数据映射到高维空间的数学函数。如果对数据的关系有明确的多项式假设，用多项式核。如果数据分布复杂或不确定关系，优先选择高斯核（RBF 核）。如果问题适合神经网络风格的映射，或者需要更简单的模型尝试，可以用 Sigmoid 核。

$$L = \max_{\alpha_i \geq 0} \left\{ \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{N} t^{(i)} t^{(j)} \alpha_i \alpha_j (x^{(i)})^T x^{(j)} \right\} \qquad \text{subject to: } \alpha_i \geq 0, \ \sum_{i=1}^{N} \alpha_i t^{(i)} = 0$$

$$\min_{w,b,\xi} \frac{1}{2}\|w\|^2 + \lambda \sum_{i=1}^{N} \xi_i$$
$$\text{subject to} \quad \xi_i \geq 0; \quad t^{(i)}(w^T x^{(i)} + b) \geq 1 - \xi_i, \quad \forall i = 1, \ldots, N$$

SVM 如何处理线性不可分的数据?
对于线性不可分的数据，SVM 使用核函数将数据映射到高维空间，使其在高维空间中线性可分。例如，使用 RBF 核函数将数据从低维空间非线性映射到高维。

SVM 的目标函数是什么？

1. Describe the classification status of this data point in the following situation:

(1)$\alpha_i = 0$ Answer: (1) Correctly classified, not a support vector

(2)$0 < \alpha_i < \lambda$ (2) Correctly classified, support vector

(3)$\alpha_i = \lambda$ (3) Support vector or misclassified

2. For both regression problem and classification problem, what components are necessary for learning?

1. Data
2. Feature Engineering
3. Model
4. Loss Function
5. Regularization Techniques
6. Learning Algorithm(gradient descent)

Get any

什么是软间隔（Soft Margin）？为什么需要软间隔?
软间隔允许部分数据点越过超平面或落在错误的区域中，从而在分类性能和模型的泛化能力之间取得平衡。通过引入松弛变量和惩罚参数 CCC，可以控制模型的容错能力和复杂性。

1. Compare the main differences between Bagging and Boosting in terms of training process and results.

   Answer:

Bagging:

- Each model is trained independently and the training data is generated by Bootstrap sampling.
- The final result is obtained by averaging (regression) or voting (classification).
- Mainly used to reduce variance and solve overfitting problems.

Boosting:

- The weak learner is trained sequentially, and each iteration adjusts the weights according to the wrong samples in the previous round, focusing on the samples that are difficult to classify.
- The final result is obtained by weighted combination.
- It is mainly used to reduce bias and improve model performance.

2. What is the difference between a linear autocoder and a linear regression?

Answer:

Linear Autoencoder:
- Unsupervised;
- Input: x, Output: x
- Objective: Minimize reconstruction error
- Purpose: Dimensionality reduction or feature extraction

Linear Regression:
- Supervised
- Input: x, Output: y
- Objective: Minimize prediction error
- Purpose: Prediction of target variable

8. How does the decision boundary look like for all above models used for classification

1. Logistic Regression
•Decision Boundary: Linear
2. Linear Regression (used for classification)
•Decision Boundary: Linear
3. Decision Trees
•Decision Boundary: Piecewise Linear and Axis-aligned
4. K-Nearest Neighbors (KNN)
•Decision Boundary: piecewise-linear, boundaries of Voronoi partition and Data-dependent
5. Perceptron
•Decision Boundary: Linear
6. Multilayer Perceptron (MLP)
•Decision Boundary: Highly Non-linear and Complex

1. What are the differences between regression problem and classification problem?

•**Regression**: The objective is to predict a **continuous output**. This involves estimating a mapping function (f) from input variables (X) to a continuous output variable (Y). Regression is used to understand relationships between variables and for predicting values within a continuous range.
•**Classification**: The objective is to predict a **categorical output**. The task is to approximate a mapping function (f) from input variables (X) to discrete output variables (Y). The goal is to identify which category or class the input data belongs to.

题目: 为什么集成方法可以提高分类性能?

答案: 集成方法可以提高分类性能的原因主要有两个：

1. **方差减少**：如果训练集是独立的，平均多个分类器可以减少方差，而不影响偏差。
2. **偏差减少**：对于简单的模型，多个模型的平均可以增加模型的容量，从而减少偏差。

4.各模型的输出值是什么?

答案:

模型	输出值类型	输出值范围	应用场景
Linear Regression	连续值	无固定范围	回归
Logistic Regression	概率值	[0, 1]	二分类
MLP	概率/连续值	视任务而定	分类/回归
Nearest Neighbors	类别标签/连续值	数据范围/离散值	分类/回归
Decision Trees	类别标签/连续值	数据范围/离散值	分类/回归
Multi-class Classification	概率/类别标签	[0, 1] 或离散值	多分类
Clustering	簇标签/簇中心	离散值/连续值	聚类
PCA	主成分值	无固定范围	降维/特征提取
SVM	间隔值/类别标签	视任务而定	分类/回归
Ensemble Methods	概率/类别标签/连续值	视基模型而定	分类/回归

题目: 什么是混合专家模型（Mixture of Experts）？

答案: 混合专家模型是一种集成方法，其中每个专家（分类器）的权重不是固定的，而是根据输入样本动态调整。通过一个门控网络（Gating Network）来决定每个专家对当前样本的贡献程度，从而实现专家之间的分工合作。

Q2. Why do we use the softmax function in multi-class problems?
A2: Softmax turns the model's output into probabilities for each class, making sure they add up to 1.

Q3. How does k-means clustering work, and what is its problem?
A3: k-means groups data into k clusters based on distance to centers. It doesn't work well with odd-shaped or uneven-sized groups.

Q4. Why do neural networks need activation functions?
A4: Activation functions make the network non-linear, so it can learn complex patterns in the data.

Q5. How does the learning rate affect the training of a neural network?
A5: A high learning rate may cause the model to overshoot the optimal solution, while a low learning rate can make training slow and may get stuck in local minima.

Q: Let $X = \{\text{Raining, Not Raining}\}$, $Y = \{\text{Cloudy, Not Cloudy}\}$,

1. Calculate the entropy of a joint distribution $H(X,Y)$;
2. Calculate the entropy of Y given that it is rain
3. Calculate the expected conditional entropy.

	Cloudy	Not Cloudy
Raining	$\frac{24}{100}$	$\frac{1}{100}$
Not Raining	$\frac{25}{100}$	$\frac{50}{100}$

$$H(X,Y) = -\sum_{x \in X} \sum_{y \in Y} p(x,y) \log_2 p(x,y) = 1.56$$

$$H(Y|X=x) = -\sum_{y \in Y} p(y|x) log_2 p(y|x) = -\frac{24}{25} \log_2 \frac{24}{25} - \frac{1}{25} \log_2 \frac{1}{25} = 0.24$$

$$H(Y|X) = \sum_{x \in X} p(x) H(Y|X=x) = 0.75$$

SVM和EM的默认标签为-1和1，其余的都为0和1